



Sindar
AÑO 1

REVISTA
MENSUAL
COLECCIONABLE



P.V.P.

395

IVA inc.

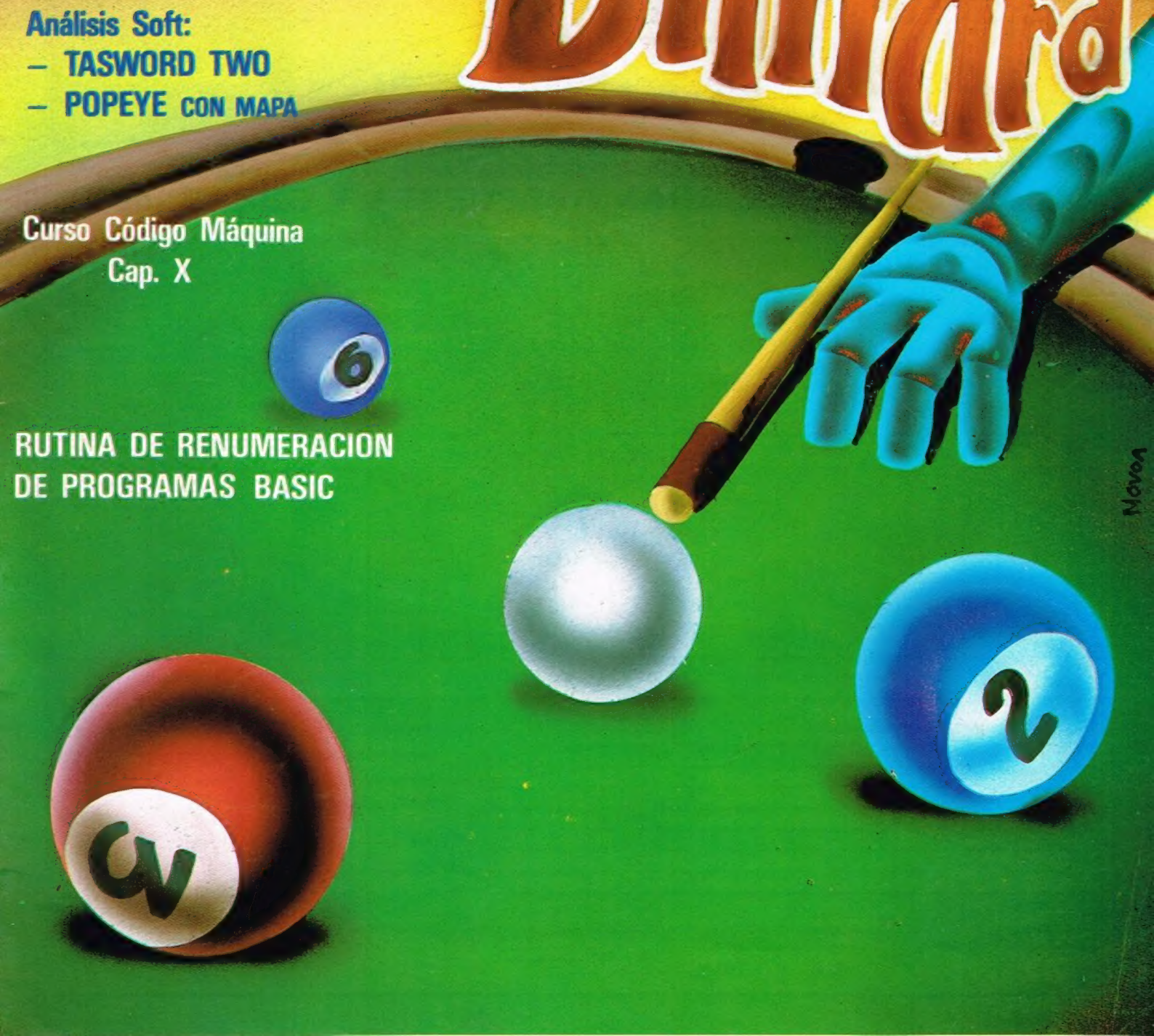
AMERICAN Billiard

Análisis Soft:

- TASWORD TWO
- POPEYE CON MAPA

Curso Código Máquina
Cap. X

RUTINA DE RENUMERACION
DE PROGRAMAS BASIC



Novon



MONSER, S.A.
Calle Argos, 9
28037 MADRID
Telf.: 742 72 12/96

Ahora Vd. puede tener todo su equipo de ordenador en un gabinete de estilo con tres elegantes niveles. No más desórdenes de cables ni de periféricos. Además su equipo estará más protegido.

NO PIERDA ESTA OPORTUNIDAD UNICA

Tendrá espacio a su alcance para hardware y software.

Dispondrá de una unidad de puente de 56,5 cm ancho, 17 cm de alto y 30,5 cm de fondo para su televisor o monitor.

Debajo de esta unidad hay espacio suficiente para guardar su ordenador, aparato de cassette o microdrive.

En una tercera unidad tiene amplio espacio para guardar cintas, diskettes, joysticks, revistas, libros, etc.

Se vende desarmado en una caja plana, es muy fácil de armar, utilizando solamente una llave ALLEN.

El gabinete se presenta en dos colores, NOGAL y ROBLE y tiene dimensiones que se ajustan a las necesidades de espacio y altura que Vd. requiere.

ANCHO 85,5 cm. • ALTO 79,5 cm. • FONDO 60 cm.

Y ADEMÁS LOS INTERFACES PARA SU JOYSTICK, IMPRESORA O MICRODRIVE

Interfáz DK'Tronics
Doble salida en la parte superior. La primera para joystick tipo Kempston y la segunda para software con teclas 6, 7, 8, 9 y 0 o redefinición de teclas.
Ref. 30001. P.V.P. 3.760 ptas.

Interface Centronic.
Para impresora y microdrive en paralelo. Ref. 30010.
P.V.P. 11.358 ptas.

Para pedidos simplemente rellene el cupón.

ORDENA TU ORDENADOR

**8.975
ptas.**



LOS JOYSTICKS DE GRAN RESPUESTA Y DURABILIDAD



**1.900
ptas.**

REF. 30007



**2.598
ptas.**

REF. 30003



**3.724
ptas.**

REF. 30008



**2.450
ptas.**

REF. 30005

Por favor, envíenme:(marco con una X):

	REF.	COLOR	CANTIDAD	PRECIO	SUBTOTAL
GABINETE	40005	Nogal		8.975	
	40005	Roble		8.975	
JOYSTICK	30007			1.900	
	30003			2.598	
	30008			3.724	
	30005			2.450	
INTERFACE	30001			3.760	
	30010			11.358	

Gastos de envío gabinete 800 ptas., joysticks 200 ptas.

TOTAL

Forma de pago:

- ☐ Talón bancario a nombre de MONSER, S.A.
☐ Giro postal núm.
☐ Contra reembolso

Nombre

Domicilio

Ciudad..... Provincia

C.P. Telf.:

FIRMA



AÑO 1 - N.º 10 - 1985 -

DIRECTOR
José Nieto Rubio

COORDINADOR
Félix Santamaría Avila

SUPERVISOR SOFTWARE
Gustavo Cano Muñoz

DISEÑO
Marco Antonio Díaz

**REDACCION
Y
COLABORADORES**

Victoria Aguilar
Javier González

PORTADA
Mauro Novoa

EDITA
MONSER, S. A.

DIRECTOR EDITORIAL
J. L. Cano Regidor

**REDACCION,
ADMINISTRACION
Y PUBLICIDAD**
Argos, 9
28037-MADRID
Tel. 742 72 12/96

**PUBLICIDAD Y
SUSCRIPTORES**
Yolanda Bardillo

FOTOCOMPOSICION
CRISOL, S. A.
Virgen del Val, 48

FOTOMECANICA
IMAGEN

IMPRIME
GRAFICAS IBARRA

DISTRIBUCION
R. B. / .

Depósito legal: M-10.328-1985
Reservados todos los derechos.
Se solicitará control O.J.D.

Sumario:

4 Análisis Software. Comentario:
Popeye
Tasword two

8 Código máquina. Cap. X

10 Juego del mes. American billards

16 Rutina de renumeración

18 Tablón

Recorta o copia

¡¡SUSCRIBETE A 48K!!

De regalo recibirás el n.º 1 del popular SOFTWARE MAGAZINE que incluye 2 fabulosas cassettes con programas estrella más la revista «Super Juegos».

Solicito me inscriban como suscriptor de su revista por un año (12 entregas). 4.740 ptas.

A partir del número..... inclusive

El importe lo abonaré de la siguiente forma:

- ☐ Giro postal n.º
☐ Contra reembolso
☐ Talón bancario a MONSER, S.A. C/ Argos, 9. 28037-MADRID

Nombre

Dirección

Ciudad D.P.

Telf.: Provincia

ANÁLISIS

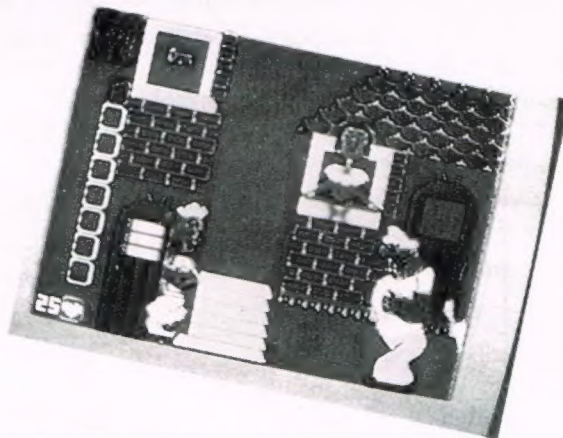
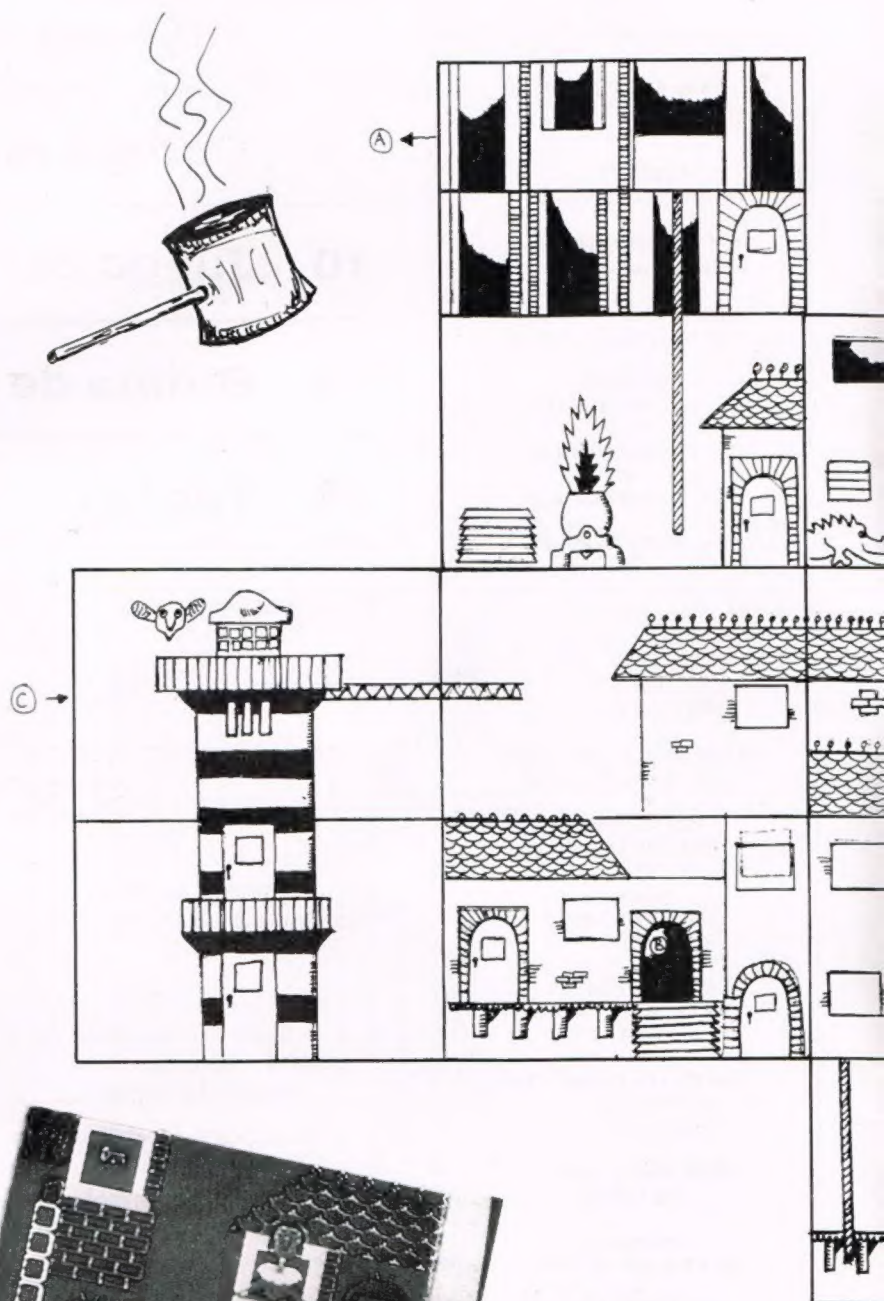
COMENTARIO: POPEYE

Nos encontramos frente a un programa en el cual los gráficos se han tratado con gran originalidad. El juego, como últimamente se ha puesto de moda, es de un famoso personaje de dibujos animados, «Ojo saltado», o para entendernos, Popeye.

El juego consiste en obtener 25 corazones distribuidos por los distintos escenarios y llevárselos a Olivia, la huesuda compañera de Popeye y sin los cuales perderemos su amor y por lo tanto, el juego. Asimismo, nos encontramos a algunos de los personajes del comic que nos harán la vida imposible, a saber: Brutus, la Bruja; el Buitre, etc. Hay que alejarse de ellos, ya que con un simple contacto harán que Popeye muerda el polvo.

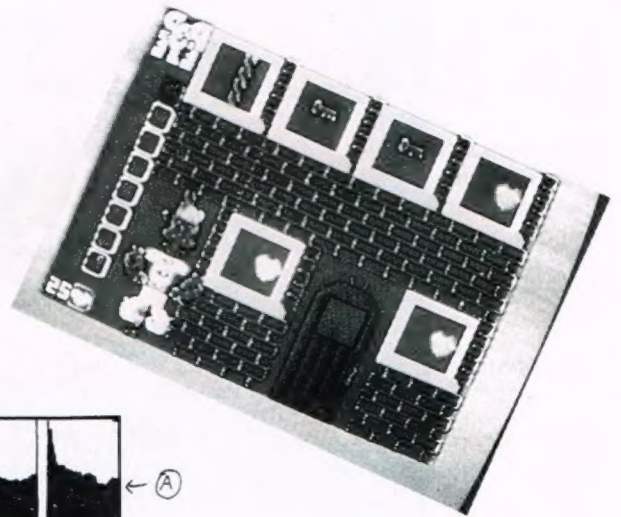
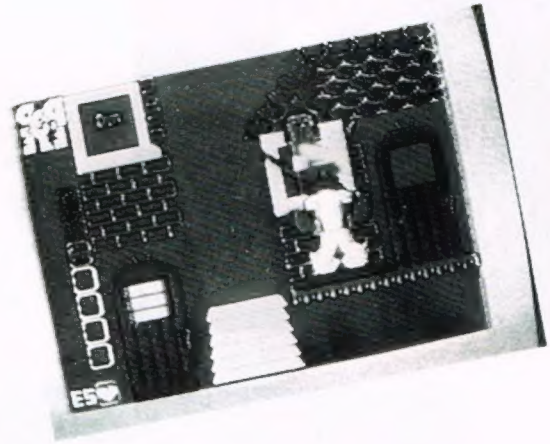
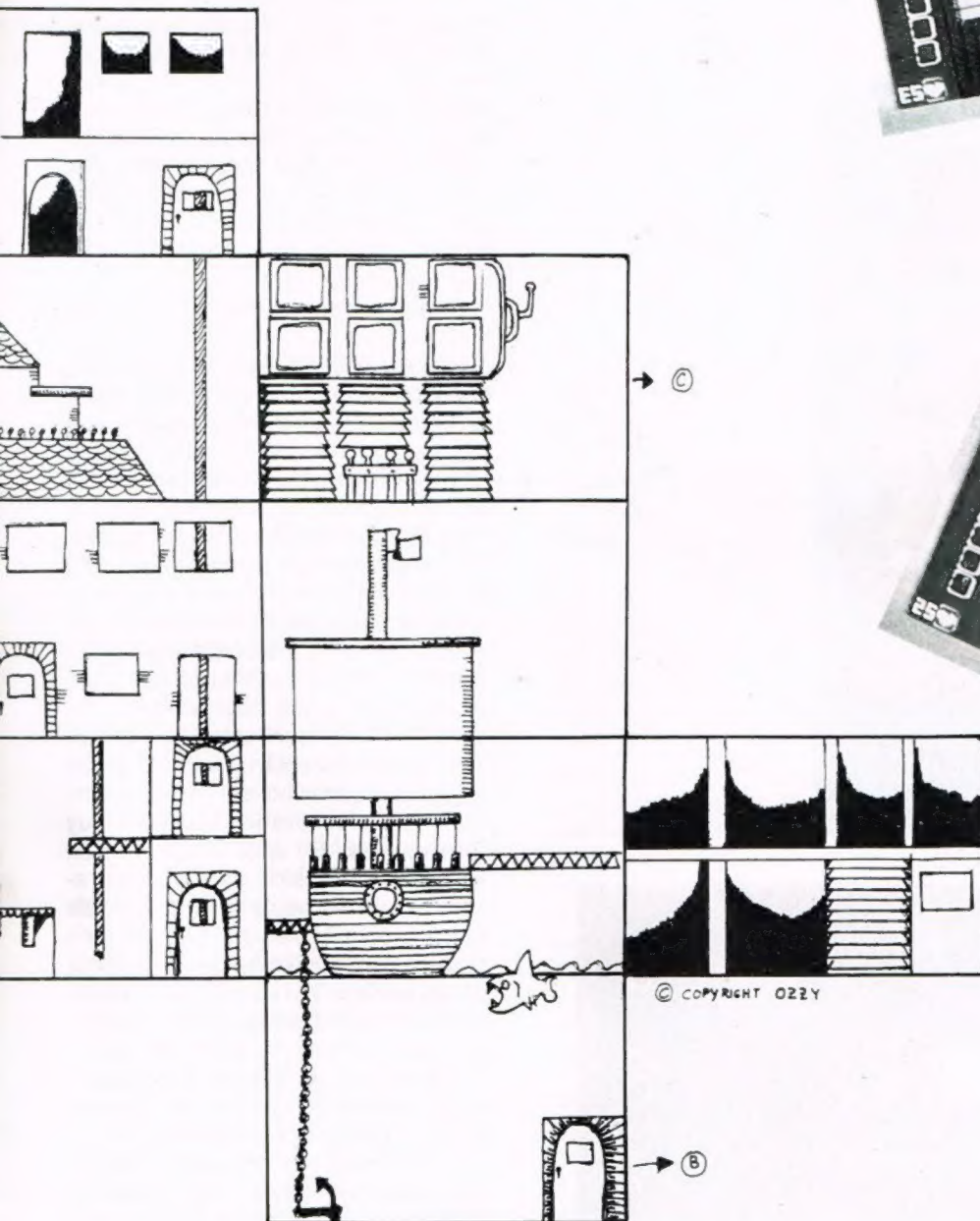
Pero hablaremos ahora más lentamente de la misión de Popeye. El juego se desarrolla en una ciudad portuaria (con faro y todo). En las habitaciones, tejados, sótanos, etc., encontraremos 3 elementos fundamentales: corazones, espinacas y llaves.

Los corazones, como ya hemos dicho, tienen que ser recogidos y llevados a la casa de Olivia (a la derecha del faro, primera ventana). Como observarás, hay corazones que no se pueden coger, ya que algunas puertas nos lo impiden. Pues bien, aquí entran en juego las llaves. Debes recolectar llaves con las que puedas abrir habitaciones y obtener corazones. Como ejemplo podemos decir que si cogemos la llave que se encuentra en la puerta de la cuerda, podremos abrir la puerta del faro, pudiendo así subir a la parte superior del faro, obtener otra llave y abrir la puerta del piso intermedio del faro para así obtener un corazón y un bote de espinacas.



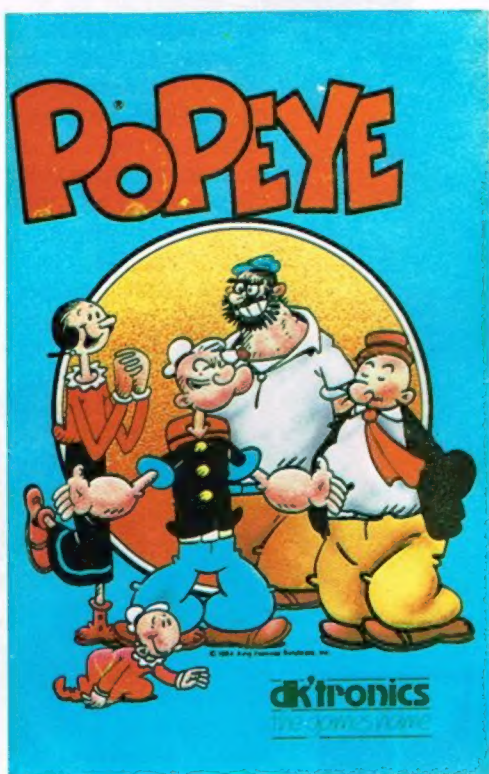
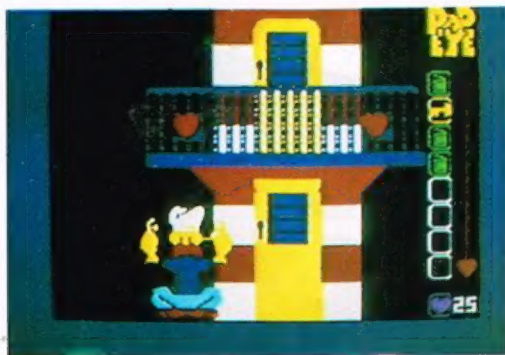
SOFTWARE

MAPA DE POPEYE®



© COPYRIGHT OZZY

ANÁLISIS



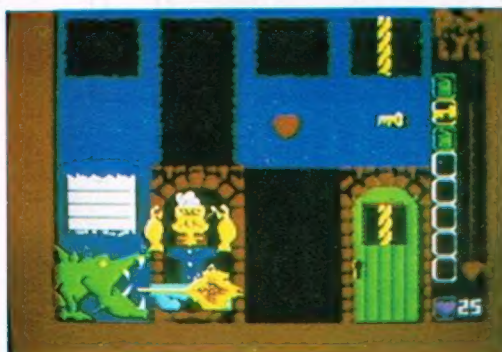
Decíamos que el segundo objeto importante eran las espinacas. Estas te serán de gran utilidad cada vez que tengas un encontronazo con algún personaje. Cuando ya no tengas más reservas de espinacas y te «sacudan», se acabará el juego. Por lo tanto haz el mayor acopio posible de ellas. Aparte de estos objetos básicos, encontrarás otros que de una u otra forma te ayudarán a completar tu misión... serás tú el que deberás encontrar su función.

Debes tener en cuenta que sólo podrás llevar encima un total de 8 objetos, y podrás saber lo que tienes observando los 8 cuadrados que hay a la derecha de la pantalla. A la derecha de estos cuadrados está lo que podríamos denominar «amorometro», es decir, el aparato que nos indica el nivel de amor de Olivia hacia Popeye. Sabes que cuando esté bajo, deberás llevarle corazones a Olivia o perderás.

Hablaremos ahora del tratamiento de los gráficos. Lo primero que observamos es que las figuras son más grandes de lo que nos tenían acostumbrados los fabricantes de software. Los colores están muy bien elegidos y los escenarios, así como los personajes y sus movimientos están muy conseguidos. En este juego existen «capas» o zonas de pantalla. Podremos movernos hacia delante y hacia atrás (aunque parece que no nos movemos) pudiéndonos esconder detrás del faro, así como ir por laberintos con profundidad. Muy original, aunque difícil de entender al principio.

Adjunto a este artículo te damos el mapa de Popeye que sin duda te ayudará a conseguir corazones.

En definitiva, un juego muy original, con buenos gráficos y muy adictivo. Un gran acierto de la casa Dk' Tronics.



SOFTWARE

TASWORD-TWO

ANÁLISIS DE SOFTWARE
PARA 48K DICIEMBRE
=====

Lo cierto es que cuando cargué el TASWORD-TWO desde mi microdive aún no se me había ocurrido el tema sobre el que escribiría en el siguiente número de 48K. Ya empezaba a estar un poco harto de tanto jueguecito como pasaba por mis manos y a pesar de ser el que esto escribe, un auténtico adicto de los videojuegos, también de cuando en cuando, necesita un respiro. Otro de los motivos que me impulsaron a cambiar el tema sobre el que tanta tinta ha corrido, era el de pensar que también habrá alguien entre nuestros lectores, que quiera o necesite dar a su ordenador otro uso que el trivial divertimento matamarcianil.

Lo cierto es que cuando cargué el TASWORD-TWO desde mi microdive aún no se me había ocurrido el tema sobre el que escribiría en el siguiente número de 48K. Ya empezaba a estar un poco harto de tanto jueguecito como pasaba por mis manos y a pesar de ser el que esto escribe, un auténtico adicto de los videojuegos, también de cuando en cuando, necesita un respiro. Otro de los motivos que me impulsaron a cambiar el tema sobre el que tanta tinta ha corrido, era el de pensar que también habrá alguien entre nuestros lectores, que quiera o necesite dar a su ordenador otro uso que el trivial divertimento matamarcianil.

Dado el precio al que algunos de nosotros adquirimos en su día el maravilloso SPECTRUM, no sería rentable, si por nuestro trabajo necesitásemos una máquina de escribir, tener que comprarla, y muchísimo menos si tenemos en cuenta que la máquina de escribir más barata que yo conozco, cuesta alrededor de las 14.000 pesetas. Ese es uno de los motivos, quizá el primero, al empezar a trabajar con un procesador de textos. La verdad sea dicha, ese

sólido argumento se nos viene abajo desde el primer momento en el que empezamos a profundizar en las posibilidades que nos ofrece cualquier procesador de textos y en particular el TASWORD-TOW.

Reconozco que es una casualidad, pero se da el caso de que mi mujer es por profesión, secretaria y para más detalles, de dirección, lo que nos da idea de la presentabilidad que deben tener sus escritos. Pues bien, como decía, fue por casualidad que un día, mientras le contaba las excelencias de este procesador, cosa que hago, con gran aburrimiento por su parte, con todo programa que llega a mis manos, noté cómo ella comenzaba a interesarse por el TASWORD-TWO, hasta el punto de rogarme encarecidamente que lo cargase un momento en la máquina para que ella pudiera dar crédito a sus oídos.

Tras algunos minutos de prueba, se volvió hacia mí, y me dijo: «Chapeau..., por fin descubro que tu maquinita roba-horas, sirve para algo más que para hacerme pasar las tardes en casa».

Ya comenté antes que ella es secretaria, lo que no comenté es que en su puesto de trabajo dispone de una máquina sólo y exclusivamente dedicada a «procesar textos», y aunque la máquina en cuestión seguro que sabe hacer muchas más cosas, nadie se las pide, pues el Software para cualquier ordenador profesional es excesivamente caro.

«¡¡Pero si es exactamente igual que el mio...!!», exclamó.

Y es cierto, el TASWORD-TWW es un procesador perfectamente llamado profesional. Con él podemos escribir, centrar, mover, copiar, buscar, tabular, salvar, cargar, margear, ir al principio, ir al final; borrar líneas, palabras, letras, y, lo más importante, imprimir los resultados.

Y para más prestaciones, que en un procesador muchísimo más caro tiene que hacerlo un experto, aquí lo haces tú mismo sin necesitar ningún conocimiento de informática, estoy hablando del formateo del buffer de la impresora.

Esto es formidable.

COMANDO

MAQUINA

CAPITULO X

1.—INTRODUCCION

En este capítulo vamos a contemplar las instrucciones de comparación y búsqueda dentro del juego de instrucciones del Z-80. Incluiremos también una instrucción que en cierto modo incluye una comparación y que se utiliza frecuentemente para formar bucles.

Si quisiéramos realizar una analogía entre las instrucciones que trataremos en este capítulo y las del lenguaje Basic que pudiéramos hacerlas corresponder, diríamos que en este capítulo contemplaríamos los bucles FOR... NEXT y los IF, principalmente. Sin embargo, para explicar las instrucciones de búsqueda que vamos a ver, en Basic tendríamos que recurrir a crearnos rutinas compuestas por bucles y comparaciones. No obstante hemos de señalar que las búsquedas mediante una instrucción en lenguaje Assembler están limitadas a encontrar un sólo carácter.

2.—Instrucciones de comparación

El formato de estas instrucciones en lenguaje Assembler es:

CP r

en donde,

r representa a alguno de los registros simples, A, B, C, D, H, y L, o a una constante cuyo valor está comprendido entre 0 y 225 en decimal (#00 - #FF).

En la siguiente tabla podemos ver el conjunto de instrucciones de comparación y su correspondiente traducción a Código Máquina.

Assembler	Código máquina
CP A	BF
CP B	B8
CP C	B9
CP D	BA
CP E	BB
CP H	BC
CP L	BD
CP n	FE xx

Como puede apreciarse, todas estas instrucciones ocupan un solo byte de memoria, excepto la última, que ocupa dos, por tener que llevar el valor a comparar de un modo directo.

La función de estas instrucciones es la de realizar una comparación entre el operando que aparece en la instrucción (registro simple o constante) y el contenido del registro Acumulador.

La forma de llevar a cabo esta comparación es por medio de una resta. Pero no hay motivos para asustarse porque el contenido de todos los operandos, tanto el Acumulador como el registro simple, no se modifican. La sustracción, que es del tipo A-r, se realiza en la Unidad Lógica Aritmética del microprocesador y lo único que sufre modificación, cara al programa, son los indicadores o flags.

En efecto, según cual sea el resultado de esta resta, los indicadores o flags se modificarán de la siguiente manera:

C Indica que:

El carry flag se activa si el número a comparar, contenido en el registro simple, o la constante, es mayor que el contenido del Acumulador. Al

hacerse la resta se produciría un desbordamiento (overflow) y el bit C se activaría.

Si C = 1 es que $r > A$

Si C = 0 es que $r \leq A$

Z Indica que:

El bit de cero se activa si el resultado de la resta es cero. Esto indicaría indudablemente que ambos operandos son iguales.

Si Z = 1 $r = A$

Si Z = 0 $r < >$

P/V Indica que:

Este flag se pone «1» si no existe acarreo negativo desde el cuarto bit, también llamado de arrastre decimal. Ahora puede parecer un tanto confusa esta explicación. Sólo pedimos un poco de paciencia pues cuando veamos las instrucciones decimales notaremos su interés.

S Indica que:

Este bit de signo indica por su estado, activado o no, si el contenido del registro simple, o constante, son menores, o no, que el contenido del Acumulador. En efecto, si el resultado de la resta $A - r$ es negativo, este bit se activa y por lo tanto indica que *r* es mayor que A. Por el contrario, si A es igual o mayor que *r*, el bit se desactiva.

N Indica que:

El indicador de suma/resta se pone activado siempre que se ejecuta esta operación, porque de hecho se trata de una resta.

Más de uno se habrá preguntado el motivo por el cual existe una instrucción tan tonta como la de comparar al registro Acumulador consigo mismo (CP A). Si volvemos a revisar el estado en que quedan los flags, tal vez podamos deducir que, lo único que hace esta instrucción, es dejar todos los indicadores, menos el N, a cero. Este es el uso que se le da normalmente.

Otro grupo de instrucciones de comparación es el constituido por aquellas en las que el byte a comparar con el contenido del registro Acumulador viene determinado, no ya por el contenido de un registro o una constante, sino por una dirección que viene reflejada por un registro par.

La función de dichas instrucciones es la misma que las de las explicadas anteriormente, y los flags se

activan y desactivan siguiendo las mismas reglas. Únicamente varía en la forma de obtener el byte a comparar, que en lugar de ser de un modo inmediato en este grupo se consideran el direccionamiento indirecto (por el contenido de un registro par) e indexado por el contenido de los requisitos IX e IY más un desplazamiento.

El formato de este grupo en el lenguaje Assembler es:

CP (p)

en donde,

p representa al par de registros HL, o a los registros IX ó IY con un desplazamiento *d* que hay que sumar al contenido IX ó IY para obtener la dirección real del byte a comparar.

El conjunto de estas instrucciones, así como su correspondiente traducción a Código Máquina figura en la tabla adjunta.

Assembler	Código máquina
CP (HL)	BE
CP (IX + d)	DD BE xx
CP (IY + d)	FD BE xx

En este caso, el valor de xx es el valor del desplazamiento, y como ya habíamos visto para el caso de los saltos relativos, puede tener un valor comprendido entre -128 y +127.

La longitud de cada instrucción puede deducirse de la tabla sin necesidad de que ahondemos en más detalles.

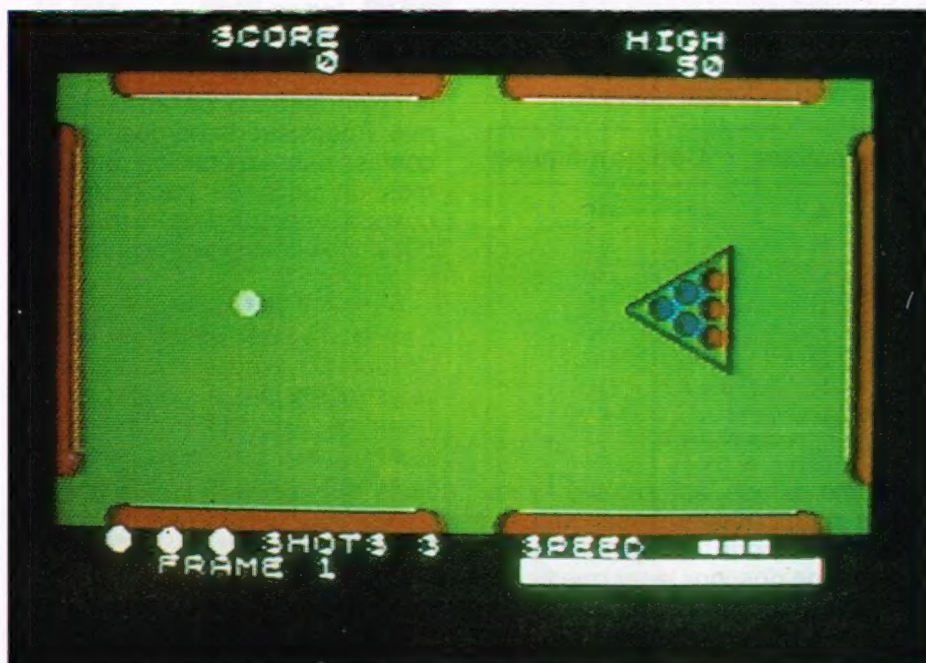
Lo normal es que tras una pregunta, y obtener su correspondiente respuesta, tomemos una decisión en función de dicha respuesta. La opción de seguir uno u otro camino, la podemos adoptar mediante la utilización de las instrucciones de bifurcación condicional, tanto absolutas como relativas, que ya hemos visto en capítulos anteriores.

Como ejercicio sechillo y fácil de llevar a la práctica, aconsejamos que se volviera a estudiar la rutina TEXTO que ya hemos visto en el anterior capítulo. Podíamos ver la posibilidad de hacer algunas sustituciones en consonancia con la función que hemos visto que realiza esta instrucción. Por si puede considerarse como una pista, aconsejamos que se repasen los motivos por los cuales se realizaba una resta y posteriormente una suma.

JUEGOS DEL MES

American Billiards

Nadie esperaba que reaccionásemos tan rápido y lo cierto es que nosotros tampoco. Apenas pasaron diez minutos desde que recibimos el aviso, cuando ya caminábamos lenta, muy lentamente, por la calle, armados tan sólo con nuestros palos, abarcando de una a otra acera. No necesitábamos pedirle a nadie que se hiciera a un lado,... ni los gatos resistían vernos de cerca y en cuanto olían nuestra proximidad se les erizaban los pelos del lomo como si de un jarro de agua fría se tratase nuestra presencia. Este lado de la ciudad era muy peligroso y más especialmente a esta hora de transición entre el día y la noche, esa hora donde todo pierde el color y el volumen para convertirse en frías siluetas que no en todos los casos se distinguen con claridad; en





momentos como este, cualquier cosa puede suceder, por eso lo más juicioso no es marchar nunca en solitario. La soledad es el peor de los males, es aburrida, sórdida, pesada y en estos casos incluso peligrosa.

A medida que avanzábamos, se oían a ras de suelo unos leves gritos y unos ojos brillantes, extraños, centelleaban en la oscuridad horizontal de la abertura de las alcantarillas; nos miraban como queriendo transmitir algo, como comunicándose solidarias con nosotros, como sintiéndonos de alguna manera hijos suyos, o al menos ahijados, pues seguramente más de una, o quizá alguna de sus abuelas, asistieron a nuestro bautizo. Una patada sobre el pavimento fue suficiente para que en una alborotada orgía de gritos, desaparecieran los cientos de pequeños ojos que ya comenzaban a rodearnos.

La calle gris desentonaba con los «graffitis» de las paredes. El arte popular, junto con el de algún entendido en bellas artes y el de algún

que otro maldito obseso, convertían cada plano vertical en una auténtica verbena en la que podrías pasar perfectamente varias horas descubriendo cosas nuevas y por lo tanto sin aburrirte ni un ápice.

Tras la última esquina nos encontrábamos en el centro de un tenebroso callejón y al fondo del mismo unas luces de neón nos anunciaban que estábamos muy próximos al garito de donde partió la amenaza. Ya hacía bastante tiempo que nos

sentíamos presos de ese desagradable hormigueo que se te pone en la nuca cuando alguien te observa sin apartar la mirada; era lógico, pues hacía más de media hora que habíamos entrado en su territorio. Cada paso que dábamos nos acercaba un poco más al epicentro del terremoto que había sido capaz de movilizar a diez de los mejores jugadores de billar de la ciudad.

Desde la puerta del oscuro antro, apenas se podían ver signos de vida en el interior del mismo, tan sólo bajo las mortecinas luces que había sobre cada mesa de billar se apreciaba el movimiento lento y pesado de cualquiera de las bolas de sebo que se pasaban una hora tras otra con el tabaco en una mano y las tizas en la otra, tanto afilando la punta como puliendo la superficie.

—¿Ya estás aquí? —te dijo con voz ronca uno de esos amasijos de carne—, eso significa que aceptas el reto, ¿no es así? —decía sin dejar de echar humo por esa boca—, el que gane podrá considerarse el rey de la ciudad. Sólo de ti depende colarlas todas, o dejar alguna para tu contrario.



3.—Decremento y salto DJNZ

Esta instrucción es una micro-rutina, lo mismo que algunas otras del microprocesador Z-80. Esto quiere decir que no realiza una sola función elemental, sino que realiza lo que un conjunto de instrucciones elementales. En efecto, esta que nos ocupa realiza un decrecimiento del contenido del registro B y si el resultado no es igual a cero realiza un salto relativo. Como puede apreciarse es una reunión de instrucciones conocidas, que podrían haberse realizado mediante una rutina en Assembler, compuesta por unas pocas órdenes. No obstante, el Z-80 nos proporciona esta rutina, que se utiliza con muchísima frecuencia, en una única instrucción con la ventaja de no ocupar más que dos bytes y además realizarse con mayor rapidez que por medio de varias instrucciones, pues la Unidad de decodificación y Control del microprocesador no interviene más que una sola vez.

El formato de esta instrucción en Assembler, así como su traducción a Código Máquina es:

Assembler	Código máquina
DJNZ d	10 xx

en donde,

d, y su equivalente, **xx**, son cifras comprendidas entre +127 y -128, rigiéndose por las mismas normas y que ya comentamos al hablar de las instrucciones de saltos relativos. Para la traducción a hexadecimal del valor **d** se utilizará también la tabla de números negativos que se adjuntó en aquella ocasión.

La ejecución de esta instrucción DJNZ se realiza en tres pasos bien definidos:

- a) El registro B se decrementa en una unidad. Este paso equivaldría a la instrucción DEC B.
- b) Se efectúa una comparación del registro B con cero. Esto equivaldría a cargar el registro Acumulador con cero y compararlo con el B. Es decir, dos instrucciones LD A,0 y CP B.
- c) Si B es igual a cero se continúa la ejecución del programa en la siguiente a la DJNZ. Por el con-

trario, si B fuera *distinto* de cero se realiza un salto relativo a la instrucción señalada por el desplazamiento **d**. Esta fase correspondería a una instrucción JR NZ, d.

Aun a costa de parecer pesados por insistir tanto, no queremos dejar pasar la ocasión de señalar dos cosas:

- 1.º Si cuando el programa llega a ejecutar esta instrucción el registro B tiene un cero, la bifurcación, o salto al desplazamiento se produce, porque si a cero le restamos una unidad el resultado es -1, diferente de cero.
- 2.º El valor de **d** es un desplazamiento relativo. Es decir, un número que hay que *sumar* o *restar* al valor que en ese momento tenga el Pc (program counter) y que en el momento de hacer la suma, el Pc apunta precisamente a la instrucción siguiente a la DJNZ.
En otras palabras, la instrucción DJNZ 0 no realiza salto alguno sea cual sea el resultado de la comparación de B con cero.

4.—Comparar e incrementar CPI

El formato de esta instrucción en Assembler es

CPI

y su traducción a Código Máquina es \neq ED A1

La función de esta instrucción es la de comparar el contenido del registro Acumulador con el byte señalado por el registro par HL, y dentro de un bloque cuya longitud está contenida en el par BC. La comparación se realiza en la siguiente secuencia de operaciones:

- 1.º El byte direccionado por el par de registros HL es comparado con el contenido del Acumulador. De acuerdo con el resultado de la comparación son activados, y desactivados, los flags Z, S, y H.
- 2.º El contenido del par HL es incrementado en una unidad.

2.º El contenido del par BC es decrementado en una unidad. En este momento, si el contenido del Acumulador y el byte direccionado por el par HL son iguales el indicador Z se activa, y en caso contrario, se desactiva. El indicador de P/V se pone a cero si el par BC es igual a cero y se pone a 1 en caso contrario.

Durante la ejecución de esta instrucción el flag Carry no se altera.

Esta instrucción, que también pertenece al grupo de las micro-rutinas puede utilizarse para buscar un byte determinado, o para ir analizando, byte a byte, un bloque comparándolo con un valor determinado.

Debemos llamar la atención sobre un pequeño detalle: tras la ejecución de la rutina, el par HL apunta al carácter siguiente a comparar y no sobre el recién comparado.

5.—Comparar y decrementar CPD

Complementaria de la instrucción que hemos visto en el párrafo anterior, el Z-80 nos proporciona otra instrucción micro-rutina que permite realizar la comparación del contenido del requisito Acumulador con el byte direccionado por el contenido del registro par HL, y todo ello dentro de un bloque cuya longitud está contenida en el registro par BC. La única diferencia con respecto a la instrucción que vimos en el anterior párrafo, la CPI, es que en lugar de incrementar el contenido del par HL, lo que esta instrucción hace es decrementarlo, o sea, restarle una unidad.

El formato en Assembler de esta instrucción es

CPD

y su traducción a Código Máquina es \neq EDA9.

La ejecución de todas las operaciones que conlleva la instrucción se realiza en la siguiente secuencia:

- 1.º El byte situado en la dirección indicada por el contenido del par HL es comparado con el contenido del registro Acumulador. Los flags Z, S, y H son coloca-

dos de acuerdo con el resultado de la operación.

- 2.º El contenido del registro par HL es decrementado en una unidad.
- 3.º El contenido del par BC es también decrementado en una unidad. Si el contenido de este registro es igual a cero el indicador P/V se pone a cero, pero si no fuera así, este indicador saldrá de esta instrucción con un uno.

Habrà que tener en cuenta que, tras la comparación, el par HL ha sido decrementado, por lo que si se ha producido la igualdad ha sido con el byte posterior al señalado por HL.

6.—Comparar incrementar y repetir CPIR

Para la búsqueda de un valor dentro de un bloque, el juego de instrucciones del Z-80 dispone también de una instrucción concreta, perteneciente al grupo de las que llamamos micro-rutinas, pues realiza una serie de funciones en una sola instrucción.

El formato Assembler de esta instrucción es

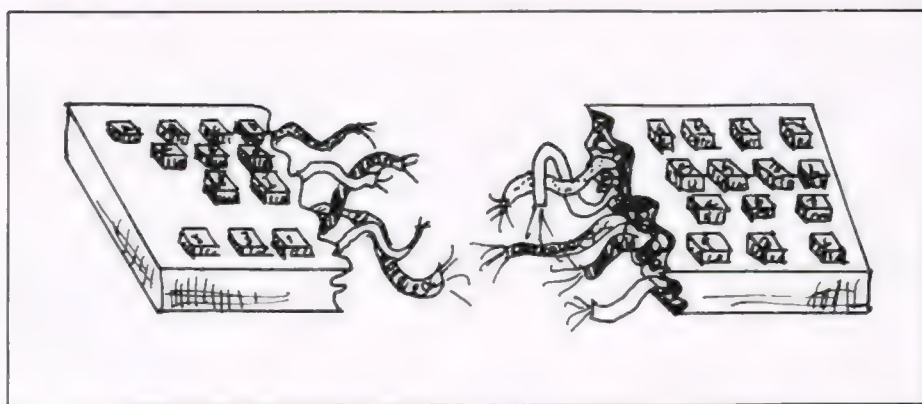
CPIR

y su traducción a Código Máquina es ≠ EDB1.

Como ya hemos indicado, esta instrucción realiza la búsqueda de un byte cuyo contenido sea igual al del contenido del registro Acumulador, dentro de un bloque de datos cuyo comienzo está contenido en el par HL, en el momento de iniciar la ejecución de la instrucción, y cuya longitud está contenida en el registro par BC.

La secuencia de operaciones que se realizan al ejecutar esta instrucción es la siguiente:

- 1.º El byte direccionado por el contenido del registro par HL es comparado con el contenido del Acumulador. Los indicadores Z, S y H son actualizados de acuerdo con el resultado de la operación.
- 2.º El registro HL es incrementado en una unidad.
- 3.º El registro BC es decrementado en una unidad.



- 4.º Si el par BC es igual a cero, el indicador P/V es puesto a uno.
 - 5.º Si la comparación realizada en el primer paso no ha detectado igualdad, y el contenido del registro par BC no es igual a cero, se volverá a la ejecución del paso 1.º.
- Si la comparación refleja igualdad, o el registro par BC es cero, el control del programa pasa a la instrucción siguiente a la CPIR.

Cuando termina de ejecutarse esta instrucción ya sabemos que ha sucedido una de estas dos condiciones: o no hay un carácter igual al contenido en todo el bloque (indicador P/V activado, y el flag Z desactivado), o en efecto ha encontrado un carácter igual (flag Z activado, P/V desactivado). Con tal de analizar el contenido del flag Z, mediante una instrucción de salto condicional, podremos diferenciar cuál de las dos condiciones se ha cumplido.

7.—Comparar, decrementar y repetir CPDR

También para la instrucción de Comparar, Incrementar y Repetir existe una complementaria en la que en lugar de incrementar se decrementa. Podríamos decir que, mientras la anterior instrucción buscaba el primer byte que dentro de un bloque era igual al contenido del Acumulador, con esta instrucción que nos ocupa se busca el último byte, dentro del bloque, cuyo contenido es igual al del registro Acumulador.

El formato Assembler de esta instrucción es

CPDR

y su traducción a Código Máquina es ≠ EDB9.

De un modo «académico» podíamos decir que la función de esta instrucción es la de buscar un byte cuyo contenido sea igual al del registro Acumulador, dentro de un bloque cuyo final está direccionado por el registro par HL y cuya longitud está contenida en el registro par BC.

La secuencia de operaciones que se realizan al ejecutar esta instrucción es la siguiente:

- 1.º El byte direccionado por el contenido del registro par HL es comparado con el contenido del Acumulador. Los indicadores Z, S y H son actualizados de acuerdo con el resultado de la operación.
- 2.º El registro par HL es decrementado en una unidad.
- 3.º El registro BC es decrementado en una unidad.
- 4.º Si el par BC es igual a cero el indicador P/V es puesto a uno.
- 5.º Si la comparación realizada en el primer paso no ha detectado igualdad, y el contenido del registro par BC no es igual a cero, se volverá a la ejecución del paso 1.º.

Si la comparación refleja igualdad, o el registro par BC es cero, el control del programa pasa a la instrucción siguiente a la CPDR.

Cuando termina de ejecutarse esta instrucción ya sabemos que puede deberse a dos circunstancias: o no hay un carácter igual en todo el bloque (indicador P/V activado y el indicador Z desactivado), o en efecto se ha encontrado un carácter igual (indicador Z activado y P/V desactivado). Para analizar el resultado bastará decidir, mediante una bifurcación condicional, a partir del indicador Z.

MONSER cada día +

**6 SuperCassettes
Full Memory
por solo 1.795 pts.**



*Libreta que contiene manual
de instrucciones en Castellano
+ Aplicaciones para tu Spectrum*



*Ya a la venta en Kioscos,
Tiendas Especializadas y
Departamento de
Informatica del*

El Corte Inglés

MONSER S. A.
C/ Argos, 9 - 28037 Madrid .

Teléfonos: (91) 742 72 12 - 742 72 96

MONSER

TYPE AND RUN

LA PRIMERA REVISTA CON CASSETTE PARA TECLEAR Y GRABAR

AÑO I - N.º 7

LISTADOS PARA
SPECTRUM

Rutina de
renumeración
Matemáticas
comerciales
Lunar

Navegar

COM

allas
Super boxeo

AMSTRAD

Policía de carretera

Monser

MENSUAL

Pídala en su kiosko

o en tiendas especializadas

SUSCRIPCIONES: MONSER, S.A. C/ Argos, 9 - 28037 MADRID

LA MEJOR REVISTA CON CASSETTE



RUTINA DE RENUMERACION

Un buen programador no es aquel al que los programas le funcionan a la primera, o al menos no el mejor de todos; para mi forma de ver, un buen programador es aquel al que aunque un programa le funcione, no para hasta que contenga todos los detalles y mejoras que se le vayan ocurriendo a lo largo de varios días y... pasa lo que pasa, que llega un momento en que ya no es posible insertar ni una sola línea de programa más, pues ya hemos cubierto absolutamente todos los números que habíamos previsto en nuestro intervalo inicial; ha llegado el momento de reorganizar nuestro programa, pero no vamos a recorrerlos instrucción por instrucción cambiándolas de número, sería absolutamente absurdo sobre todo si tenemos en cuenta que lo que hay delante de nuestras narices es un «ordenador», (sería de tontos ordenar a mano).

La rutina que a continuación se explica nos ayudará en esta empresa, agotadora si no dispusiéramos de ella.

En primer lugar teclearemos el listado número uno, que como su nombre indica, es el cargador de rutinas. A continuación daremos RUN y el programa nos pedirá la dirección donde queremos colocar nuestra rutina, que para más señas es totalmente reubicable, es decir, que la podremos poner donde queramos siempre y cuando no pisemos el área de Basic ocupada por nuestro programa; tras esta primera pregunta, el ordenador nos pedirá datos, que no son otros que los que aparecen en el listado número dos y que se introducirán en sentido horizontal, línea a línea y no columna a columna como podría parecer. Cuando ya hayamos terminado de introducir los valores del listado número dos, daremos un valor cualquiera superior a 255 con lo que nos saldremos del programa cargador. En este momento os recomendamos que salvéis los códigos introducidos con el comando:

SAVE «rutinam» CODE XXXXX, 690 siendo XXXXX la dirección donde decidimos al principio del proceso colocar nuestra rutina.

Ahora teclearemos el programa que aparece con el nombre de Listado tres, y cargaremos con MERGE el programa de renumerar, y con un RUN 9990 ejecutaremos el programa tres y contestaremos a todas sus preguntas, lo que nos llevará a la renumeración automática de nuestro programa Basic.

Esta rutina renumera no sólo las instrucciones, sino también todos los GOTO's, GOSUB's, RESTORE's, etc... salvo los que son operativos, que nos los informará la propia rutina por pantalla, con el número de instrucción y lugar ocupado en la misma. Un operativo es el obtenido por medio de la aplicación de fórmulas algorítmicas, como por ejemplo GOSUB 100*N.

Listado 1

```
1 REM Cargador de rutinas
2 INPUT "donde empezamos? ";d
3 INPUT "introduce valor ";b
4 IF b>255 THEN STOP
5 FOR e=dss:byte
6 LET dss=dss+1
7 GO TO 3
```

Listado 2

005	107	13	40	110
006	000	400	100	000
009	000	100	000	000
010	000	000	000	000
014	000	000	000	000
017	000	000	000	000
047	000	000	000	000
100	000	000	000	000
200	100	000	000	000
214	100	000	000	000
217	000	000	000	000
247	000	000	000	000
250	100	000	000	000
252	000	000	000	000
254	000	000	000	000
255	000	000	000	000
256	000	000	000	000
257	000	000	000	000
258	000	000	000	000
259	000	000	000	000
260	000	000	000	000
261	000	000	000	000
262	000	000	000	000
263	000	000	000	000
264	000	000	000	000
265	000	000	000	000
266	000	000	000	000
267	000	000	000	000
268	000	000	000	000
269	000	000	000	000
270	000	000	000	000
271	000	000	000	000
272	000	000	000	000
273	000	000	000	000
274	000	000	000	000
275	000	000	000	000
276	000	000	000	000
277	000	000	000	000
278	000	000	000	000
279	000	000	000	000
280	000	000	000	000
281	000	000	000	000
282	000	000	000	000
283	000	000	000	000
284	000	000	000	000
285	000	000	000	000
286	000	000	000	000
287	000	000	000	000
288	000	000	000	000
289	000	000	000	000
290	000	000	000	000
291	000	000	000	000
292	000	000	000	000
293	000	000	000	000
294	000	000	000	000
295	000	000	000	000
296	000	000	000	000
297	000	000	000	000
298	000	000	000	000
299	000	000	000	000
300	000	000	000	000

DE PROGRAMAS BASIC

192	62	10	35	35
194	35	35	35	35
25	24	241	35	43
45	237	75	1035	22
205	43	45	2055	233
55	56	27	235	433
55	237	75	1035	22
205	43	45	2055	233
55	56	12	235	152
205	215	152	45	33
40	197	22	33	12
55	34	123	22	1
12	92	167	42	34
5	237	91	103	103
2	33	34	35	22
7	33	92	35	35
3	34	175	22	34
5	22	42	123	22
7	91	52	92	35
3	52	22	175	22
71	22	231	25	3
13	254	53	40	243
254	13	202	173	197
254	234	202	233	197
254	235	45	35	24
237	40	35	254	22
40	31	254	240	40
19	254	225	40	15
254	247	22	55	234
243	40	35	195	162
197	10	0	10	0
205	119	0	254	35
205	54	205	133	24
254	13	202	173	197
54	53	40	134	244
59	40	35	254	43
40	31	205	119	0
24	231	205	119	0
254	34	40	3	35
24	7	205	119	0
254	34	32	242	205
119	0	254	202	40
2	24	191	205	119
0	34	23	234	13
202	173	197	234	53
40	131	254	33	40
235	205	132	24	40
35	254	43	55	13
254	53	43	35	205
119	0	254	33	40
243	24	234	202	202
255	50	140	202	205
142	22	123	54	255
32	243	223	103	205
1	22	437	75	205
22	205	43	455	205
227	45	52	33	16
297	75	71	22	205
0	205	43	455	205
227	45	52	13	16
225	195	162	197	254
13	40	5	254	53
43	2	24	195	43
43	70	43	73	43
43	43	34	93	22
213	205	34	197	24

[illegible]

Listado de

```

0000 REM prepara rutina
0001 INPUT "donde esta la rutina ";indice
0002 INPUT "dame el paso entre i";paso
0003 INPUT "primer numero? ";comienzo
0004 POKE (indice+256),paso
0005 POKE (indice+256),comienzo-256*INT (comienzo/256):POKE (indice+256),INT (comienzo/256)
0006 RANDOMIZE USR indice

```


TABLON

Se precisan distribuidores en todo el territorio nacional para productos de primeras marcas de:

- Computer
- Vídeo
- Audio

Dirigirse a:

MONSER, S.A.

C/ Argos, 9

28037-Madrid

Telf. 742 72 12/96 (Srta. Yolanda)

CLUB DE USUARIOS MONSER

Si ya tienes tu ordenador (SPECTRUM, MSX, C-64) y deseas estar periódicamente informado de los nuevos programas y periféricos, así como recibir todas las ofertas de software y hardware que, con frecuencia hacemos especialmente para nuestros socios, inscríbete hoy mismo sin ningún gasto, ni compromiso por tu parte.

Enviar el cupón a:

CLUB DE USUARIOS MONSER
Calle Argos, 9. 28037-MADRID

Podrás obtener más información a través de las revistas «48K», «Type and Run», «MSX Soft Magazine» y «Commodore Soft Magazine», o llamando al teléfono (91) 742 72 12/96.

Deseo inscribirme en el CLUB DE USUARIOS MONSER sin ningún gasto, ni compromiso, para recibir información y tener acceso a los descuentos y ofertas especiales del Club.

Nombre
Dirección..... Teléfono
Ciudad..... C.P. Provincia
Edad..... Profesión
Marca del ordenador..... Fecha compra

FIRMA

FECHA

Envía inmediatamente el cupón y recibirás en breve un keypanel de regalo para personalizar tus programas.

QUEREMOS VER TUS PROGRAMAS BASIC: SPECTRUM, C64, MSX Y AMSTRAD

Sabemos que eres lo suficientemente ingenioso para no necesitar copiarlos de nadie. Mándanos una cinta de cassette con tu programa, y a ser posible, un listado del mismo.

Premiamos con 5.000 ptas. y un ejemplar de la revista en cuestión, cada programa que publiquemos en cualquiera de nuestras cinco publicaciones.

Envíanos el tuyo, cuanto antes lo hagas más posibilidades tendrás de verlo publicado.

MONSER, S.A.
C/ Argos, 9
28037-MADRID

Deprisa, deprisa...

C-30 CASSETTE ESPECIAL PARA ORDENADOR

La mini cargante

530 CASSETTE
ESPECIAL PARA
ORDENADOR

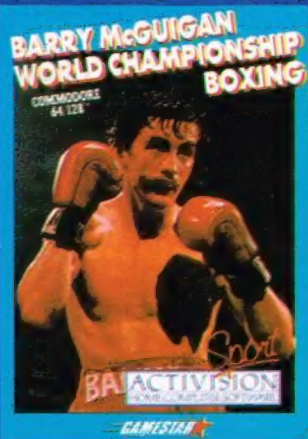
MONSER



Con la marca Monser sobre el cassette, usted obtiene no solamente una excelente cinta para computador, sino también una cassette que le proporciona todas las funciones y conveniencias que requiera el usuario. El cassette para ordenadores personales Monser está diseñado para ser usado con microcomputadores y provee una combinación única de precisión y ejecución.

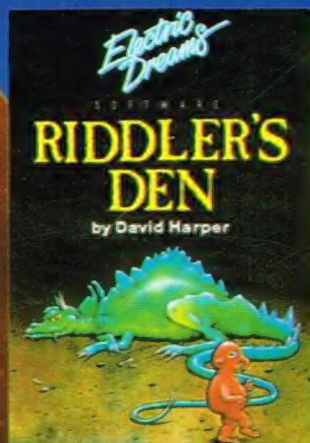
DE VENTA EN TIENDAS ESPECIALIZADAS.

Para envíos dirigirse a Monser, S.A. c/ Argos nº 9. Tlf. 742 72 12 / 96.



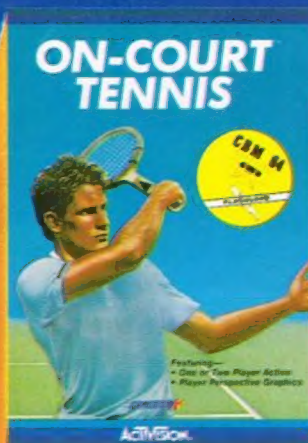
Juego en el que puedes crear a tu propio boxeador. Elige su raza, estilo físico e imagen. Entrena y demuestra sus habilidades.

C.S.A.



TRUNKIE, el hombre elefante y tu superarás las trampas y lograrás encontrar a GREGOGO, el Gran Dios de Oro.

C.S.A.



Elige la superficie de la pista y el oponente y demuestra tu control de la raqueta jugando a dobles o individual.

C.S.A.



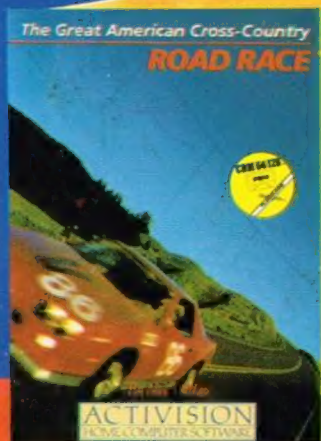
Entra en el mundo ciclista a través de las 16 etapas del Tour. Con acompañamiento musical y el jersey amarillo esperando al ganador.

C.

ACTIVISION INC.

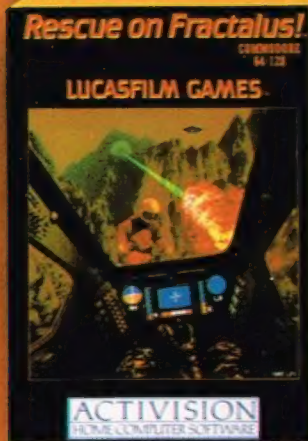
HOME COMPUTER SOFTWARE

*P.V.P.
2.200



Al volante de tu coche atraviesas 25 ciudades. Seleccionas la ruta, maniobras a través del tráfico... Todo un reto de conducción automovilística.

C.A.



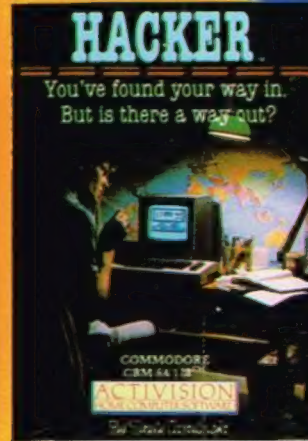
Recorre un planeta en tres dimensiones a la búsqueda de tus compañeros, mientras el enemigo te persigue.

C.S.A.



Nunca fue Aladino tan generosamente premiado por los genios. Vuele sobre una increíble alfombra mágica en tres dimensiones.

C.A.M.



Compleja aventura donde los jugadores deben buscar a través de las diferentes pistas y problemas como resolver el misterio.

C.S.A.

... y sus clásicos: GHOSTBUSTER. C.S.A.M. DEATHLON DE ACTIVISION. C.M. RIVER RAID. C.S.M.

* ENDURO. PITFALL 2 y otros títulos más a 1.540.- P.V.P. • SPACE SHUTTLE. C. DESIGNER PENCIL. C. ... etc.

Disponibles para:
COMMODORE C
SPECTRUM S
AMSTRAD A
MSX M

EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES
O DIRECTAMENTE
POR CORREO O TELEFONO A:
Argos, 9 - 28037 MADRID
Teléfs. 91/742 72 12/96

DISTRIBUIDO POR:



MONSER, S. A.